Computational Science:
Computational Methods in Engineering

The False Position Method

Outline

• Description of the Method
• Derivation of the Root Estimate
• Notes on Implementation
Description of the Method

More Intelligent “Midpoint”

False-Position Method
\[ x'_{\text{new}} = \frac{x_L + x_U}{2} \left( \frac{f(x_L) + f(x_U)}{f(x_L) - f(x_U)} \right) \]

Bisection Method
\[ x'_{\text{new}} = \frac{x_L + x_U}{2} \]

Since \( |f(x_L)| < |f(x_U)| \), it is a good assumption that the root is closer to \( x_L \) than it is to \( x_U \).
Step 1

Pick a lower and upper bound, \( x_L \) and \( x_U \) that is known to contain a single root.

![Graph showing a function and bounds](image1)

Step 2

Calculate a better estimate of the position of the root using a linear approximation.

![Graph showing a linear approximation](image2)
Step 3

Calculate the function at the new estimate for $x_r$.

Step 4

Adjust the bounds.
Step 5

Estimate the position of the root by linear approximation.

Derivation of the Root Estimate
Derivation of Estimate (1 of 3)

The equation of a line given a point \((x_0, y_0)\) and slope \(m\) is

\[
(y - y_0) = m(x - x_0)
\]

Assuming the function connecting the bounds is close to linear, the slope is approximately

\[
m \approx \frac{f(x_U) - f(x_L)}{x_U - x_L}
\]

Derivation of Estimate (2 of 3)

Choose the lower bound to be the point \((x_0, y_0)\) in the line equation.

\[
x_0 = x_L, \quad y_0 = f(x_L)
\]

\[
y - f(x_L) = \frac{f(x_U) - f(x_L)}{x_U - x_L}(x - x_L)
\]

Now estimate the position of the root by setting \(y = 0\) and solving for \(x\).

\[
0 - f(x_L) = \frac{f(x_U) - f(x_L)}{x_U - x_L}(x_r^{\text{new}} - x_L)
\]

\[
x_r^{\text{new}} = x_L - \frac{f(x_L)}{f(x_U) - f(x_L)}(x_U - x_L)
\]
Derivation of Estimate (3 of 3)

Choose the upper bound to be the point \((x_0, y_0)\) in the line equation.

\[
x_0 = x_U \\
y_0 = f(x_U)
\]

\[
y - f(x_U) = \left(\frac{f(x_U) - f(x_L)}{x_U - x_L}\right)(x - x_U)
\]

Now estimate the position of the root by setting \(y = 0\) and solving for \(x\).

\[
0 - f(x_U) = \frac{f(x_U) - f(x_L)}{x_U - x_L}(x_{\text{new}} - x_U)
\]

\[
x_{\text{new}} = x_U - \frac{f(x_U)}{f(x_U) - f(x_L)}(x_U - x_L)
\]

Interpretation of Estimate (1 of 2)

There are now two possible equations to estimate the position of the root.

\[
x_{\text{new}}^U = x_L - \frac{f(x_L)}{f(x_U) - f(x_L)}(x_U - x_L) \\
x_{\text{new}}^L = x_U - \frac{f(x_U)}{f(x_U) - f(x_L)}(x_U - x_L)
\]

Average these equations.

\[
x_{\text{new}}^L = \frac{1}{2} \left[ x_L - \frac{f(x_L)}{f(x_U) - f(x_L)}(x_U - x_L) \right] + \frac{1}{2} \left[ x_U - \frac{f(x_U)}{f(x_U) - f(x_L)}(x_U - x_L) \right]
\]

\[
x_{\text{new}}^L = \frac{x_U + x_L}{2} - \frac{f(x_U) + f(x_L)}{f(x_U) - f(x_L)} \frac{x_U - x_L}{2}
\]
Interpretation of Estimate (2 of 2)

\[ x_{r}^{\text{new}} = \frac{x_{U} + x_{L}}{2} - \frac{f(x_{U}) + f(x_{L})}{f(x_{U}) - f(x_{L})} \frac{x_{U} - x_{L}}{2} \]

- Typical bisection method equation
- A correction term to give a better estimate.

Notes on Implementation
Block Diagram of False-Position Method

Start
Choose \( x_L \) and \( x_U \).
Choose \( x_L \) and \( x_U \).
Evaluate \( f(x) \) at endpoints.
\( f_L = f(x_L) \), \( f_U = f(x_U) \)
Evaluate \( f(x) \) at endpoints.
\( f_L = f(x_L) \), \( f_U = f(x_U) \)
Estimate position of root \( x_r \)
\[ x_r = \frac{x_L + x_U}{2} - \frac{f(x_L) + f(x_U)}{f(x_L) - f(x_U)} \cdot \frac{x_L - x_U}{2} \]
Calculate new \( x_r \)
\[ x_r = \frac{x_L + x_U}{2} - \frac{f(x_L) + f(x_U)}{f(x_L) - f(x_U)} \cdot \frac{x_L - x_U}{2} \]
Calculate change.
\[ \delta x = \frac{x_r^{\text{new}} - x_r^{\text{old}}}{x_r^{\text{old}}} \]
no \( \delta x < \text{tol} \) yes \( \delta x < \text{tol} \) \( \text{Done!} \)

Move upper bound.
\( x_U = x_r \) \( f_U = f_r \)
Move lower bound.
\( x_L = x_r \) \( f_L = f_r \)
Done!

This is the only difference from bisection method.

When False-Position Fails

The false-position method can fail or exhibit extremely slow convergence when the function is highly nonlinear between the bounds.

This happens because the estimated root is a linear fit and a very poor estimate of a nonlinear function.
Notes on False-Position Method

• Very similar to bisection method
• Calculates a more intelligent “midpoint”
• Converges much faster for near linear functions
• Converges slower for highly nonlinear functions